VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

WEBOVÁ APLIKACE PRO PODPORU VÝUKY BIOINFORMATIKY - GENETICKÝ KÓD

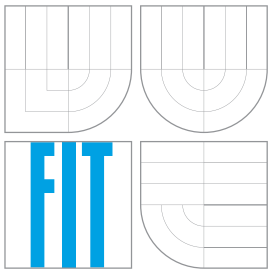
BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

MARTIN KILIÁN

BRNO 2011

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INFORMAČNÍCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INFORMATION SYSTEMS

WEBOVÁ APLIKACE PRO PODPORU VÝUKY BIOINFORMATIKY - GENETICKÝ KÓD

WEB APPLICATION FOR BIOINFORMATICS EDUCATION - GENETIC CODE

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

MARTIN KILIÁN

VEDOUCÍ PRÁCE
SUPERVISOR

Ing. IVANA BURGETOVÁ, Ph.D.

BRNO 2011

Abstrakt

V této práci jsou stručně popsány základní poznatky z molekulární biologie. Práce sa zabýva převodem informace z DNA přes RNA do proteinu, taktéž jsou stručně popsány metody a algoritmy zabývající se detekcí genů. Větší část práce se zabýva rozpoznáváním startovacích signálů a určovaním kódujících sekvencí a jejich následnou transkripcí a translací, čehož demonstrační program bylo potřeba vytvořit. Výsledkem je webová aplikace jako výukový systém.

Abstract

This thesis briefly describes basic informations in moleculary biology. Thesis deals with a transfer of information from DNA into RNA through uprotein, are also briefly described the methods and algorithms dealing with the detection of genome. The larger part deals with the recognition cue coded signals and determination of sequences and their subsequent transcription and translation, which demonstration program was needed to create. The final result is s web application as the education system.

Klíčová slova

proteosyntéza, translace, transkripce, prokaryota, dna, rna, protein

Keywords

proteosynthesis, translation, trancription, prokaryotes, dna, rna, protein

Citace

Martin Kilián: Webová aplikace pro podporu výuky
bioinformatiky - genetický kód, bakalářská práce, Brno, FIT VUT v Brně, 2011

Webová aplikace pro podporu výuky bioinformatiky - genetický kód

Prohlášení

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pani Ing. Ivany Burgetovej Ph.D

.....
Martin Kilián
16. mája 2011

Poděkování

Ďakujem pani Ing. Ivane Burgetovej Ph.D za ochotu pri poskytovaní odborných konzultácií a za rady, ktoré boli nápomocné pri vypracovávaní bakalárskej práce. Slečne Lucii Benedikovej za podrobné vysvetlenie nejasností týkajúcich sa problematiky molekulárnej biológie.

Obsah

Kapitola 1

Úvod

Témou tejto bakalárskej práce je zaujímavá problematika prenosu informácií z DNA do proteínov cez RNA pomocou genetického kódu. Táto problematika sa začleňuje do oblasti bioinformatiky. Jedná sa teda o biologický problém, ktorý si bezpodmienečne vyžaduje použitie informačných technologií, algoritmov a matematických metód. Cieľom práce je vyvinúť webovú aplikáciu ako výukový program schopný demonštrácie nálezu proteínových sekvencií pre ľubovoľné DNA alebo RNA a opačne. Riešenie sa zameriava na prácu s jednoduchými štruktúrnymi génmi prokaryotických buniek. Na prvý pohľad jednoduchá úloha avšak po hlbšom preskúmaní všetkých súvislostí obsahuje niekoľko netriviálných problémov.

Kapitola 2 obsahuje teoretické informácie z bionformatiky a základné poznatky z chémie, ktoré sú potrebné pre pochopenie danej problematiky. Zaoberám sa tu základnými pojmami z molekulárnej biológií a bližšie popisujem oblasti využité pri vypracovávaní praktickej časti práce.

Ďalej v kapitole 3 rozoberám problematiku predikcie génov, kde popisujem metódy a algoritmy používané v praxi, porovnávam prokaryotá a eukaryotá a venujem pozornosť detekcii promotoru.

V 4-tej kapitole popisujem technické riešenie projektu, použité technológie a nástroje a vývojové prostredie.

Kapitola 5 je venovaná implementácii kde rozoberám ako som postupoval pri praktickej časti práce na jednotlivých podúlohach, je tu popísaná aj adresárová štruktúra a uživatelské rozhranie.

Posledná kapitola je venovaná záverečnému zhodnoteniu práce.

Kapitola 2

DNA, RNA a proteosyntéza

Táto kapitola vysvetľuje termíny z molekulárnej biológie, približuje termíny a pojmy potrebné pre pochopenie problematiky prenosu informácií z DNA.

2.1 DNA a RNA

DNA - Deoxyribonukleová kyselina je materiálnym nosičom genetickej informácie, vyskytuje sa vo všetkých bunkových organizmoch, v ktorej je predurčené ako sa bude daný organizmus vyvíjať ako bude vyzerať a ako budú prebiehať rôzne procesy v danom organizme, jednoducho povedané v našej DNA je biologickým kódom zapísané všetko o nás a našich vlastnostiach. DNA obsahuje sacharidovou zložku (kapitola 2.1.2), dusíkaté bázy (kapitola 2.1.3) a fosfátové skupiny. Je to látka, ktorá tvorí bunkové jadro (nucleus), z toho vyplýva jej názov nukleová [7].

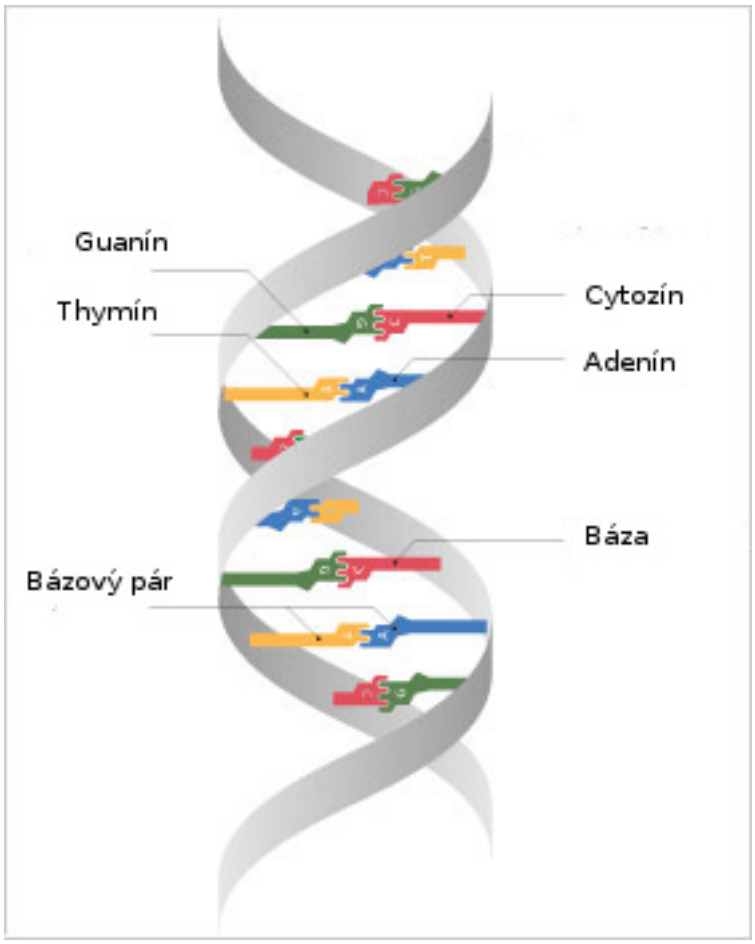
RNA - Ribonukleová kyselina je nenahraditeľná zložka proteosyntézy. Proteosyntéza je základným procesom, ktorým sa informácia obsiahnutá v DNA prevádza cez RNA do proteínu. Pod pojmom proteosyntéza môžeme rozumieť prenos genetickej informácie z génu do proteínu. Prvou časťou proteosyntézy je transkripcia nasledujúca ďalšou časťou nazývanou translácia. Je dôležité zdôrazniť, že daný popis proteosyntézy a následný popis detekcie promotorov kódujúcich oblastí je braný z pohľadu jednuduchých štruktúrných génov [7].

2.1.1 Štruktúra DNA a RNA

Molekula DNA sa skladá z dvoch polynukleotidových reťazcov, ktoré sú navzájom prepojéne vodíkovými mostíkmi. Najčastejšie sa zobrazuje ako dvojvláknová pravotočivá závitnica α helix (obrázok 2.1). Zmienený reťazec sa skladá z chemických báz. Báza je heterocyklická zlúčenina odvodená od purínu alebo pyrimidínu. Purinové bázy: adenín(A), guanín (G) a pyrimidinové bázy: cytozín (C) a thymín (T). Spolu s fosforečnanovou skupinou vytvára každá častica nukleotid, čo je základná stavebná jednotka polynukleotidového reťazca. Ako už bolo spomenuté DNA je rozdelené na dva reťazce. Jedno vlákno je nazvané ako pozitívne alebo aj pracovné, označované ako vlákno v smere 5´- 3´. Druhé vlákno je vlákno templátové, negatívne, toto vlákno je dôležite pre prenos informácie, pre replikáciu pretože od tohoto vlákna sa kopíruje informácia do mRNA. Polynukleotidové reťazce sú v DNA antiparalelné. Zastúpenie báz v DNA má svoje pravidlá. Komplementarita báz vysvetluje prečo je v každej závitnici DNA obsah molekúl purínových báz rovný počtu pyrimidínových báz. Podľa tohto pravidla má každá báza svoj komplement v paralelnom vlákne. Adenín

stojí vždy oproti thymínu a opačne. Cytozín stoji vždy oproti guanínu a opačne. Týmto vznikajú takzvané bázové páry. Podľa tohto jednoduchého pravidla je možné vždy podľa jedného vlákna dopočítať jeho komplement [7].

RNA má podobnú štrukturú ako DNA. Taktiež ju tvoria dva polynukleotidové reťazce ale namiesto deoxiribózy obsahuje ríbózu. Odlišuje sa aj výskytom pyrimidínovej bázy uracilu miesto thymínu. Reťazce sú tak isto komplementárne kde adenín stojí vždy oproti uracilu, kedže RNA postráda prítomnosť thymínu [7].



Obrázok 2.1: Molekula DNA [1]

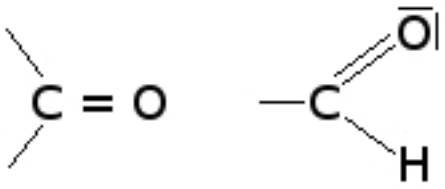
Druhy RNA dôležité pre problematiku tejto práce:

- **mRNA** - mediátorová RNA, dôležitá pre prepis informácie z DNA.
- **tRNA** - transférová RNA, dôležitá pre prenos informácie z mRNA do ribozómu.
- **rRNA** - ribozómová RNA, dôležitá pre transláciu

2.1.2 Sacharidové zložky

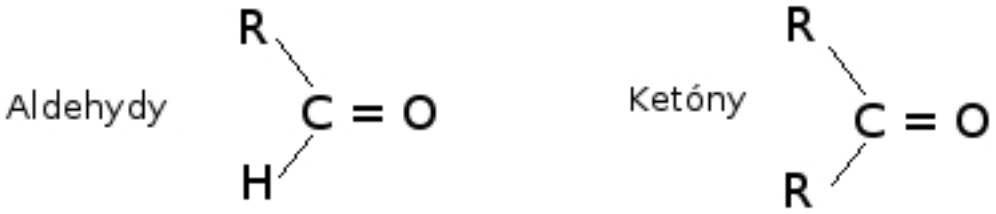
Ribonukleová kyselina RNA tak isto ako deoxyribonukleová kyselina DNA obsahuje vo svojej štruktúre sacharidovú zložku. Týmto sacharidom je v RNA ribóza (D-ribóza) a v DNA

deoxyribóza (2-deoxy-D-ribóza). Oba sacharidy patria do skupiny ALDOPENTÓZ. Pentózy preto, lebo vo svojej molekulovej štruktúre majú naviazaných 5 uhlíkov -C-C-C-C-C-. A ketózy preto, lebo vo svojich štruktúrach majú naviazanú funkčnú aldehydickú skupinu odvodenú od aldehydov (Obrázok 2.2) [12].



Obrázok 2.2: Ketónová a aldehydová skupina [12]

Aldehydy sú deriváty uhľovodíkov, ktoré obsahujú karbonylovú skupinu. Rozdelujeme ich podľa naviazaného uhľovodíkového zvyšku. Podľa toho delíme aj sacharidy na aldózy a ketózy (Obrázok 2.3) [3].



Obrázok 2.3: Aldehydy a ketóny [3]

Tieto sacharidy sú schopné otáčať rovinu polarizovateľného svetla (skúmanie polarimetrickými a refraktometrickými metódami). Hovoríme, že sú opticky aktívne. Sú pravo (+) alebo ľavo (-) točivé. Radíme ich do L a do D- radu.

- D - rad - skupina sa nachádza v štruktúre sacharidu na pravo.
- L - rad - skupina sa nachádza v štruktúre sacharidu na ľavo.

Otáčajú sa preto, lebo majú chirálne uhlíky. Chirálny uhlík je ten, na ktorom sú naviazané 4 rôzne funkčné skupiny.[12]

2.1.3 Dusíkaté bázy

Dusíkaté bázy patria medzi heterocyklické zlúčeniny, ktorých kruh obsahuje okrem uhlíka a vodíka aj iné atómy. Nazývame ich heteroatómy. Sú to najčastejšie kyslík, síra, dusík (O, S, N). Do uhľovodíkového cyklu sa môže naviazať jeden alebo viac rovnakých alebo rôznych heteroatómov [12].

Heterocyklické zlúčeniny sú základom zložitých štruktúr, ktoré sú súčasťou rastlinných a živočíšnych organizmov - sacharidov, nukleových kyselín, aminokyselín, vitamínov a alkaloidov [12].

Podľa veľkosti heterocyklu rozlišujeme:

- Päťčlenné heterocykly s jedným alebo viacerými heteroatómami.
- Šesťčlenné heterocykly s jedným alebo viacerými heteroatómami.
- Kondenzované heterocykly s jedným alebo viacerými heteroatómami.

Šesťčlenné heterocykly s dvoma heteroatómami.
Od pyrimidínovej štruktúry sa odvodzujú dusikaté bázy, ktoré sa podieľajú na stavbe nukleových kyselín už spomínané pyrimidýnové bázy (cytozín, tymín, uracil) [12] [3].

Heterocyklické zlúčeniny s dvoma kondenzovanými heterocyklami
Jeho štruktúra je odvodená od pyrimidínového a imidazolového cyklu (imidazol je päťčlenný heterocykel s dvoma heteroatómami). Je to pevná kryštalická látka zásaditého charakteru. Od jeho štruktúry sa odvodzujú dusikaté bázy, ktoré sú súčasťou štruktúry nukleových kyselín tzv. purínové bázy (adenín, guanín) [3] .

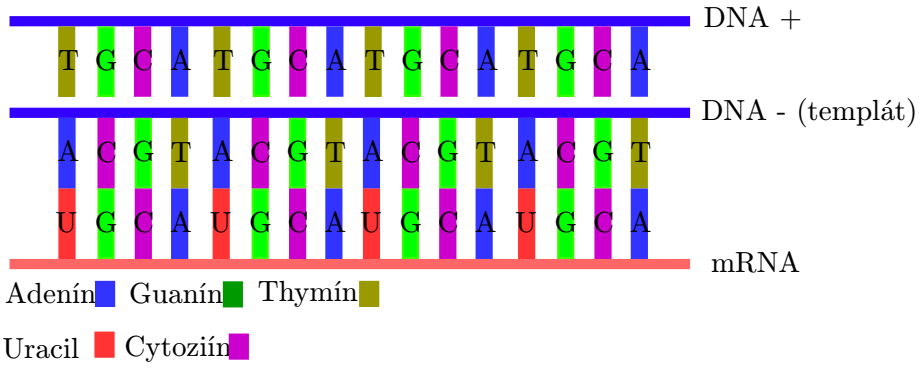
Zvyšok kyseliny trihydrogenfosforečnej spája jednotlivé pentózy (cukry) v nukleových kyselinách (DNA, RNA). Väzba medzi pentózou (cukrom) a dusíkatou bázou v nukleových kyselinách (DNA, RNA). Zásada (dusikatá báza) sa v nukleotide viaže N- glykozidovou väzbou na prvý atóm uhlíka sacharidu. Sacharid sa súčasne estericky viaže (cez svoj piaty atóm uhlíka) s kyselinou trihydrogénfosforečnou[12].

2.1.4 Transkripcia

Skôr než dôjde k vysvetleniu čo je transkripcia je dôležité vysvetliť pojem kodón. V minulosti boli vykonávané mnohé experimenty za účelom pochopenia genetického kódu. Objav kodónu majú na svedomý dvaja páni Sydney Brenner a Francis Crick, ktorý v roku 1961 objavili mutáciu posunu rámca. Prišli na to, že odstránením jednej alebo dvoch báz sa výsledná štruktúra výrazne zmení, natožto odstranením troch báz neovplyvnilo štruktúru proteínu. Vďaka tomuto objavu prišli na to, že každú aminokyselinu kóduje práve jedna trojica nukleotidov [8].

Transkripcia je prvá fáza proteosyntézy. Proces začína prepisom čiže transkripciou. Aby sa informácia obsiahnutá v DNA v géne mohla dostať do proteínu musí sa niekam prepísať. Na to slúži už zmienená mRNA. V tejto fáze sa sekvencia nukleotidov v DNA prepisuje do sekvencie mRNA. V mieste kde začína daný gén nasadá RNA-polymeráza a k jednotlivým bázam DNA priradzuje komplementárne bázy RNA(obrázok 2.4). Toto sa deje na templátovom DNA vlákne čo je vlákno negatívne v smere 3´- 5´ [7].

Pred každou sekvenciou u prokaryot, ktorá sa prepisuje z DNA do RNA sa nachádza úsek zvaný promotor. Táto sekvencia nie je prepisovaná, ale v tejto fáze na ňu nasadá RNA-polymeráza. Za promotorom sa nachádza sekvencia, ktorá podlieha transkripcii. Transkripcia je ukončená keď reťazec narazí na terminátor, čo je ukončujúca sekvencia kde sa polymeráza odpojí. Terminátoru predchádza stop kodón. Promotor a terminátor majú význam najmä pre správne napojenie a odpojenie polymerázy a ostatných molekúl, ktoré sa účastnia transkripcie. V praxi obvykle býva jeden promotor spoločný pre niekoľko za sebou idúcich génov. Tieto gény maju spoločný názov operon. To, či sa má transkripcia vykonať alebo nie, ovplyvňuje represor. Ten úzko súvisí s operátorom, čo je sekvencia,



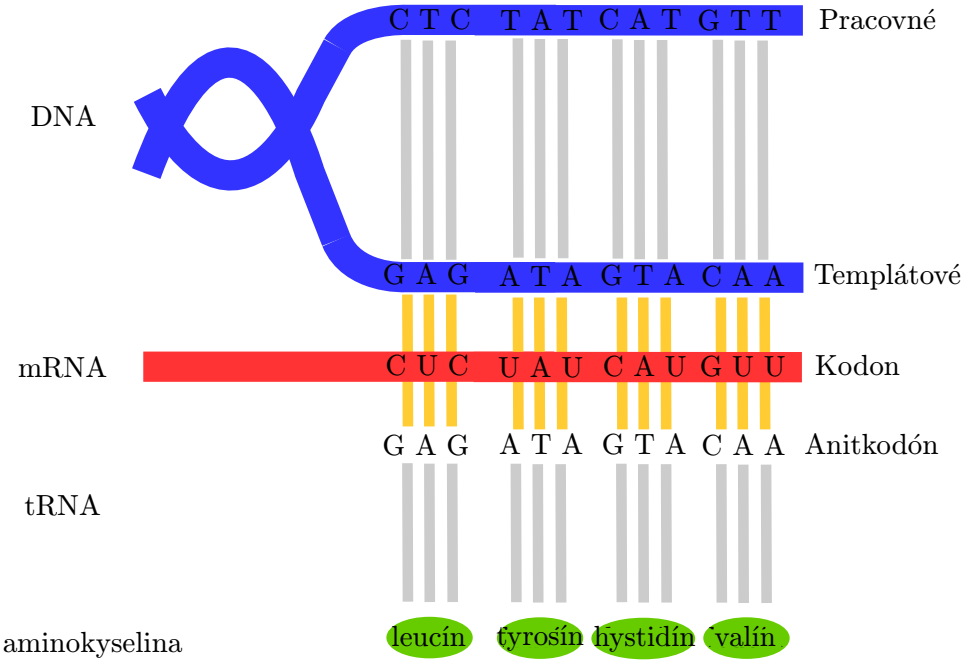
Obrázok 2.4: Schéma transkripcie

ktorá nasleduje hneď za promotorom. Tento represor ovplyvňuje to, či sa RNA polymeráza naviaže na promotor a zaháji transkripciu. V tomto procese sa nachádzajú ešte ďalšie molekuly, ktoré riadia činnosť represora. Sú to induktory a korepresory. Keď sa induktor naviaže na reprosor tak ho inaktivuje, čo spôsobí, že RNA polymeráza zaháji transkripciu. Prítomnosťou korepresora sa však represor aktivuje a zabráni RNA polymeráze v transkripcii. Transkripcia pre zložené štruktúrne gény je zložitejšia, no daná problematika nieje predmetom tejto práce [7].

2.1.5 Translácia

Translácia je druhý krok proteosyntézy. Pri translácii sú dôležité dve zložky RNA a to rRNA a tRNA. K molekule mRNA sa pripojujú ribozómy, ktoré sú tvorené molekulami rRNA. Ribozóm sám o sebe sa skladá z malej a veľkej jednotky. Pri translácii sa tieto jednotky spoja. K menšej jednotke sa pripája zmienená mRNA a potom sa pripojí veľka jednotka. Dôjde ku kontaktu mRNA a tRNA a pripojí sa aminokyselina do proteínového reťazca. Proteínový reťazec je sled výsledných aminokyselín získaných z informácií nesených v géne. Daný transport aminokyseliny do ribozómu zabezpečuje tRNA. Vlákna tRNA majú len 74 až 95 nukleotidov. Graficky zobrazená sekundárna štruktúra pripomína ďatelinový trojlístok. Vlákno je tvorené štyrmi ramenami, kde podobne ako v DNA a RNA sú komplementy spojené vodíkovými mostíkmi. Pri proteosyntéze sú dôležité akceptorové rameno a antikodónové rameno. Antikodónové rameno obsahuje trojicu nukleotidov - antikodón. Medzi antikodónom mRNA a DNA je úzka súvislosť (obrázok 2.5). Každý antikodón je komplementom k príslušnej časti mRNA, kde táto mRNA je komplementom k príslušnej DNA templátového vlákna, z čoho vyplýva že antikodón je vždy rovnaký ako spomínané vlákno DNA s tým, že miesto thymínu sa vždy nachádza uracil [7].

Ako už bolo spomenuté ku kodónom mRNA sa priradzujú jednotlivé kodóny tRNA s príslušnými aminokyselinami. To znamená, že mRNA sa číta po takzvaných tripletoch, kde jeden triplet je vlastne trojica nukleotidov. Jednotlivé triplety mRNA teda určujú konkrétne aminokyseliny a ich poradia v proteínovom reťazci. Tieto priradenia proteínov k tripletom tvoria genetický kód. Tabuľka genetického kódu obsahuje výpis všetkých možných kombinácii tripletov a výsledné aminokyseliny. V sekvencii mRNA respektíve v samotnom DNA (keďže mRNA je vytvorený komplement s nahradeným thymínom) sa nachádzajú štart a stop kodóny alebo triplety, ktoré určujú začiatok a koniec transkripcie. Sú to triplety AUG pre štart a UAA, UAG a UGA ako terminačné kodóny. Tranlácia teda



Obrázok 2.5: Vzájomný vzťah kodónu, antikodónu a aminokyseliny

začína metionínom a keď dosiahne konca polypeptidový reťazec sa odpojí. Sekvencia medzi štart a stop kodónom sa nazýva ORF (open reading frame) [7].

Genetický kód je však degenerovaný. Kombináciou všetkých nukleotidov môžeme získať až 64 tripletov, no výsledných aminokyselín je len 20 (tabuľka 2.1 a 2.2). To znamená, že jednu aminokyselinu kódujú vždy viac ako dva triplety okrem výnimočných prípadov kde napríklad metionin je kódovaný vždy len jedným trippletom, v niektorých prípadoch môže aminokyselinu kódovať aj 6 rôzych tripletov ako napríklad arginín [7].

2.2 Aminokyseliny

Sú základné stavebné jednotky bielkovín. Z chemického hľadiska sú to substitučné deriváty karboxylových kyselín, ktoré sú kyslíkaté deriváty uhlovodíkov R-COOH [3].

Pre bielkoviny sú dôležité tie aminokyseliny, ktoré majú aminoskupinu NH_2 viazanú na α-uhlíku (susediaceho s karboxylovou skupinou). Tento atóm uhlíka je chirálny (viaže 4 rôzne skupiny). Od aminokyselín sa môžu odvodiť optické izoméry s D- a L- konfiguráciou výnimku tvorí kyselina aminooctová - glycín. Príslušnosť k radu D- alebo L- sa pri aminokyselinách odvodzuje od D- a L- serínu, podobne ako sa odvodzujú optické izoméry sacharidov od D- a L- glyceraldehydu [12].

Dnes poznáme viac ako 300 aminokyselín. V bielkovinách sa vyskytuje 20 takzvaných proteinogénnych aminokyselín len v L- konfigurácii. Niektoré aminokyseliny obsahujú okrem uvedených základných charakteristických skupín aj ďalšie skupiny [12].

V biochémii sa pre aminokyseliny používajú aj triviálne aj systémové názvy. Od triviálnych sa odvodzujú trojhláskové skratky. Stavbu bielkovín zabezpečuje živočíšny organizmus

Kodón	Aminokyselina	Kodón	Aminokyselina
UUU	fenylalanín	UCU	serín
UUC	fenylalanin	UCC	serin
UUA	leucín	UCA	serín
UUG	leucín	UCG	serín
CUU	leucín	CCU	prolín
CUC	leucín	CCC	prolín
CUA	leucín	CCA	prolín
CUG	leucín	CCG	prolín
AUU	izoleucín	CCU	treolín
AUC	izoleucín	CCC	treolín
AUA	izoleucín	CCA	treolín
AUG	metionín	CCG	treolín
GUU	valín	GCU	alanín
GUC	valín	GCC	alanín
GUA	valín	GCA	alanín
GUG	valín	GCG	alanín

Tabuľka 2.1: Zoznam aminokyselín a príslušných kodónov [7]

využívaním aminokyselín z potravy alebo vlastnou biosyntézou. Aminokyseliny, ktoré daný organizmus nie je schopný syntetizovať z iných látok, sa musia nahrádzať v potrave. Nazývame ich nevyhnutné alebo esenciálne aminokyseliny. Tie, ktoré si organizmus vie syntetizovať z iných látok nazývame nahraditeľné alebo neesenciálne. Obsah esenciálnych aminokyselín v bielkovinách rozhoduje o ich biologickej hodnote [12].

Významnou úlohou aminokyselín je schopnosť navzájom sa zlučovať do väčších celkov. Pri kondenzácii aminokyselín reaguje karboxylová skupina jednej molekuly a aminoskupina nasledujúcej aminokyseliny. Pritom sa môžu viazať rovnaké alebo rozdielne aminokyseliny. Postupne sa tvoria podľa počtu viazaných aminokyselín dipeptid, tripeptid až polypeptid [12].

Kodón	Aminokyselina	Kodón	Aminokyselina
UAU	tyrosín	UGU	cysteín
UAC	tytosín	UGC	cysteín
UAA	STOP	UGA	STOP
UAG	STOP	UGG	tryptofán
CAU	histidín	CGU	arginín
CAC	histidín	CGC	arginín
CAA	glutamín	CGA	arginín
CAG	glutamín	CGG	arginín
AAU	asparagín	AGU	serín
AAC	asparagín	AGC	serín
AAA	lysín	AGA	arginín
AAG	lysín	AGG	arginín
GAU	Kyselina asparágová	GGU	glycín
GAC	Kyselina asparágová	GGC	glycín
GAA	Kyselina glutámová	GGA	glycín
GAG	Kyselina glutámová	GGG	glycín

Tabuľka 2.2: Zoznam aminokyselín a príslušných kodónov [7]

Kapitola 3

Predikcia génov

Predikcia génov patrí medzi netriviálne záležitosti. Je založená na kódujúcom potenciále, využíva toho, že v kódujúcej a nekódujúcej oblasti sa nenachádzajú rovnaké informácie, respektíve rozloženie nukleotidov v daných oblastiach je rozdielny. Je potrebné rozlišovat medzi prokaryotickými a eukryotickými génmi, na predikciu génu v každom z nich sa aplikujú rôzne na mieru upravené metódy. Veľkou nevýhodou je fakt, že potrebujeme už existujúce dáta [10].

3.1 Bioinformatika

Bioinformatika sa zaoberá metódami pre zhromažďovanie a analýzu rozsiahlých súborov biologických dát [4]. Je to veda spájajúca informatiku a biológiu do jedného celku. Jedná sa o disciplínu, ktorá využivá algoritmizáciu, matematické postupy a počítačové vedy na riešenie biologických problémov. V konečnom dôsledku ide o vytváranie počítačového softwaru na vytváranie, spracovanie a uchovávanie biologických dát alebo aj spojenie umenia programovania s biologickými znalosťami. Bionformatika je disciplínou pomerne mladou, a v našich končinách odbor nie veľmi rozšírený, ale s veľkou perspektívou, v niektorýh diskusiach je však možné sa dočítať, že bionformatika svoj svetový vrchol zažila a už nič prelomové do dnešnej spoločnosti neprinesie.

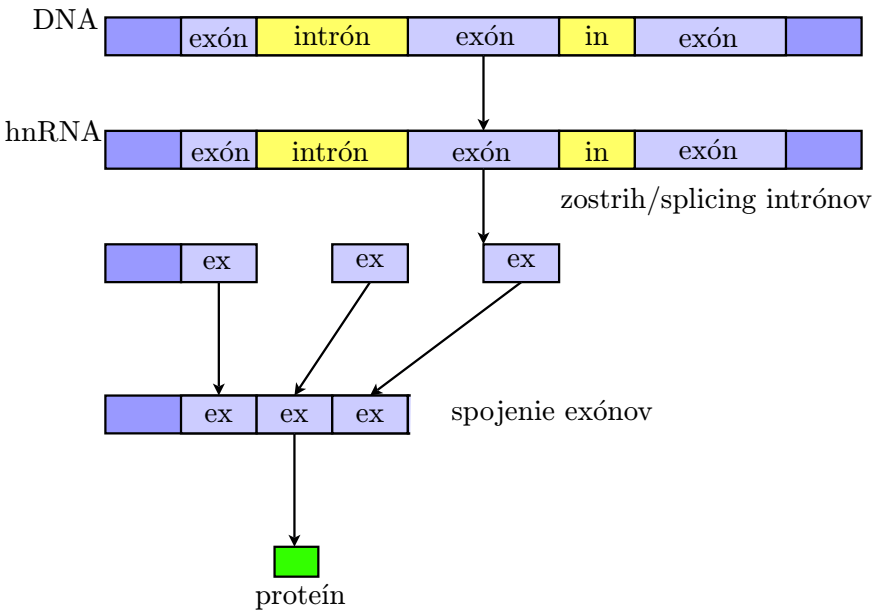
3.2 Prokaryotá vs. Eukaryotá

Medzi jednoduché štruktúrne gény patria prokaryotické gény. Prokaryotické gény sú zložené z dvoch domén archea a baktéria. Prokaryotická bunka má omnoho jednoduchšiu stavbu ako bunka eukaryotická. Sú to organizmy staré niekoľko miliárd rokov. Eukaryotické gény su zmesov intrónov a exonóv. Intrón objavili v roku 1977 páni Phillip Sharp a Richard Roberts, ktorí experimentovali s mRNA hexónu, čo je vírus. Tieto intróny sú z eukaryotických génov vystrihnuté ešte pred transláciou [8].

Úseky, ktoré v sekvencii ostanú sa nazývajú exóny, čiže intróny sú oblasti nekódujúce a exóny oblasti kódujúce. Takéto gény sa označujú ako složené gény. Odstránenie intrónov a zformovanie mRNA je složitý proces, používa sa na to metóda zostrihu(splicing) [7].

Prokaryotické bunky oproti eukaryotickým nemajú intróny a ich gény sú oveľa hustejšie 83 až 85 percent genomu obsahuje kódujúce sekvencie. U eukaryot je to len niečo okolo 5 percent. Prokaryotické gény majú spoločný promotor pre niekoľko génov, kdežto u složitejších génov je každý promotor určený prave pre jeden gén. Jednoduché gény majú 1 typ

RNA polymerázy, ktorá vykonáva transkripciu, eukaryotické majú 3 typy RNA polymeráz. Detekcia genomu v prokaryotických bunkách je jednoduchšia ako u eukaryotických už len kvôli jednoduchšej štruktúre génu. Táto práca sa zameriava práve na prokaryotické gény. Na obrázku 3.1 je zobrazený zostrih intrónov a tvorba proteínu [8].



Obrázok 3.1: Zostrih intrónov a tvorba proteínu u eukaryot

3.3 Detekcia promotoru

Dôležitou časťou tejto práce je samotná detekcia promotorov pre transkripčné operony aby bolo možné určiť z akej časti DNA sekvencie je možné urobiť prepis informácie cez RNA do proteínu. Keďže promotor sa do mRNA neprepisuje pracuje sa s vláknami samotnej DNA. Ako už bolo zmienené promotor je určitá sekvencia nukleotidov, ktorá signalizuje RNA polymeráze, že ma začat transkripciu. Určenie tejto časti DNA reťazca nieje až tak triviálne ako sa na prvý pohľad môže zdať a v praxi sa musia aplikovať složité metódy a algoritmy aby sa dosiahlo čo najpresnejších výsledkov.

Detekcia promotoru je dôležitá z hľadiska určenia kódujúcej oblasti, pretože štart a stop kodóny sa môžu nachádzať aj v nekódujúcich regiónoch, kde však ich výskyt nemusí znamenať začiatok a stop translácie. Najjednoduchšou metódou je naivná metóda, ktorá vlastne nedetekuje promotor, ale len vyhľadáva štart a stop kodóny a ORF dlhšie ako 21 nukleotidov, čo je veľmi nespoľahlivé a nepresné. Ďalšou metódou je metóda založená na štatistike kodónov. Aminokyselinu kóduje viacej tripletov, ale niektoré majú väčšiu pravdepodobnosť ako ostatné, čiže je to metóda založená na štatistike, ktorá má problémy pri určovaní kratších sekvencií [8].

V prokaryotických génoch sa nachádzajú takzvané Shine-Delgarnove sekvencie, tieto sekvencie sú umiestnené medzi štart kodónom a začiatkom transkripcie. Týchto sekvencii môže byť viac rôznych druhov, väčšinou ale obsahujú bázy AGGAGGU. Tieto sekvencie sú kvôli rozsiahlosti problematiky v reálnej aplikácii tejto práce vynechané [8].

Promotor je špecifická oblasť približne 35 báz pred začiatkom génu, začiatkom transkrip-

cie. V tejto oblasti sa na rôznych offsetoch pred začiatkom nachádzajú štartovacie signály. Promotor je zložený z rôznych boxov, ktoré majú svoje špecifické vlastnosti a umiestnenie [8].

3.3.1 Tata box, Pribnow box

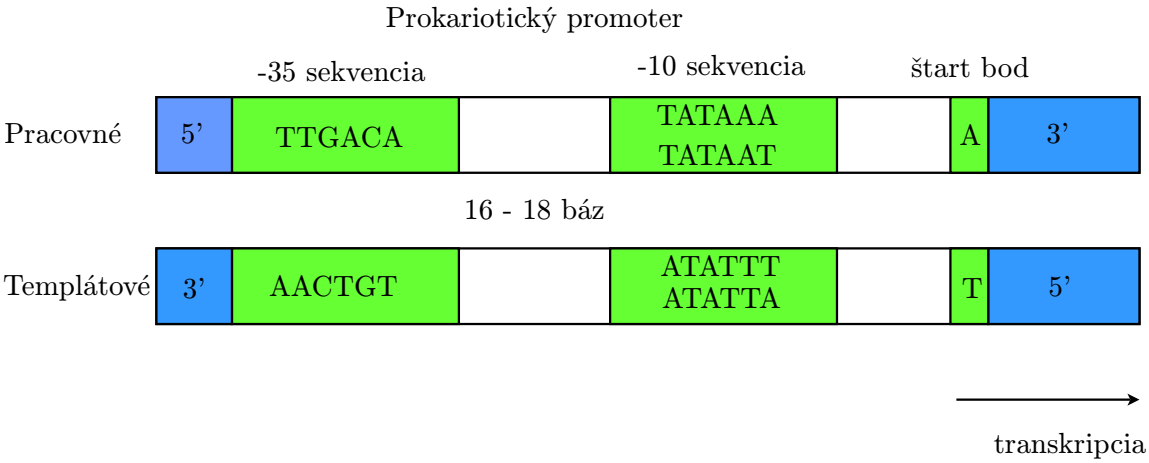
Tata box je jedným z boxov, ktoré sa nachádzajú v sekvencii promotora. Tata box je sekvencia nukleotidov, ktorá sa nachádza na -10 offsete od začiatku transkripcie (obrázok 3.2) [8].

Jedná sa o sekvenciu s obsahom báz TATAAA pre pracovné vlákno, pre templátové vlákno je to jeho komplement ATATTT, v niektorých literatúrach a zdrojoch môže byť uvedené aj ako TATAA. V reálnych pripadoch musí za tata boxom následovať štart kodón ATG vo vzdialenosti približne 150 bázi od tata boxu, v tejto práci kôli jednoduchosti a možnosti demonštrácie je braný do úvahy len rozsah 20 až 30 bázi, ak sa v tejto vzdialenosti nenachádza žiadny štart kodón daný promotor sa vyhodnotí ako nevalídny a neberie sa pri analýze reťazca do úvahy [2].

Je dôležité uvedomiť si, že tata box sa nevyskytuje len striktne v zmienenom formáte TATAAA, výskyt daných báz má určitú pravdepodobnosť. Podobnú funkciu ako tata box má Pribnow box. Tento box sa nachádza na tom istom mieste ako tata, líši sa však formátom a pravdepodobnosťou prítomných báz. Pribnow box má všeobecný formát TATAAT (ATATTA komplement).

3.3.2 Gilbertov box

Gilbertov box je ďalši box nachádzajúci sa v regióne promotora. Gilbertov box sa nachádza približne na -35 offsete od začiatku transkripcie (obrázok 3.2). Vzdialenosť medzi Gilbertovým a tata boxom je v rozsahu 16 až 18 báz. Ide o sekvenciu 6-tich nukleotidov v tvare TTGACA pre pracovné vlákno, pre templátové vlákno je to jeho komplement AACTGT. Tak isto ako pri tata boxe nemusí byť táto sekvencia v predpísanom formáte. Každá báza má svoju určitú pravdepodobnosť, takže v danej sekvencii sa môžu vyskytovať aj iné bázy.

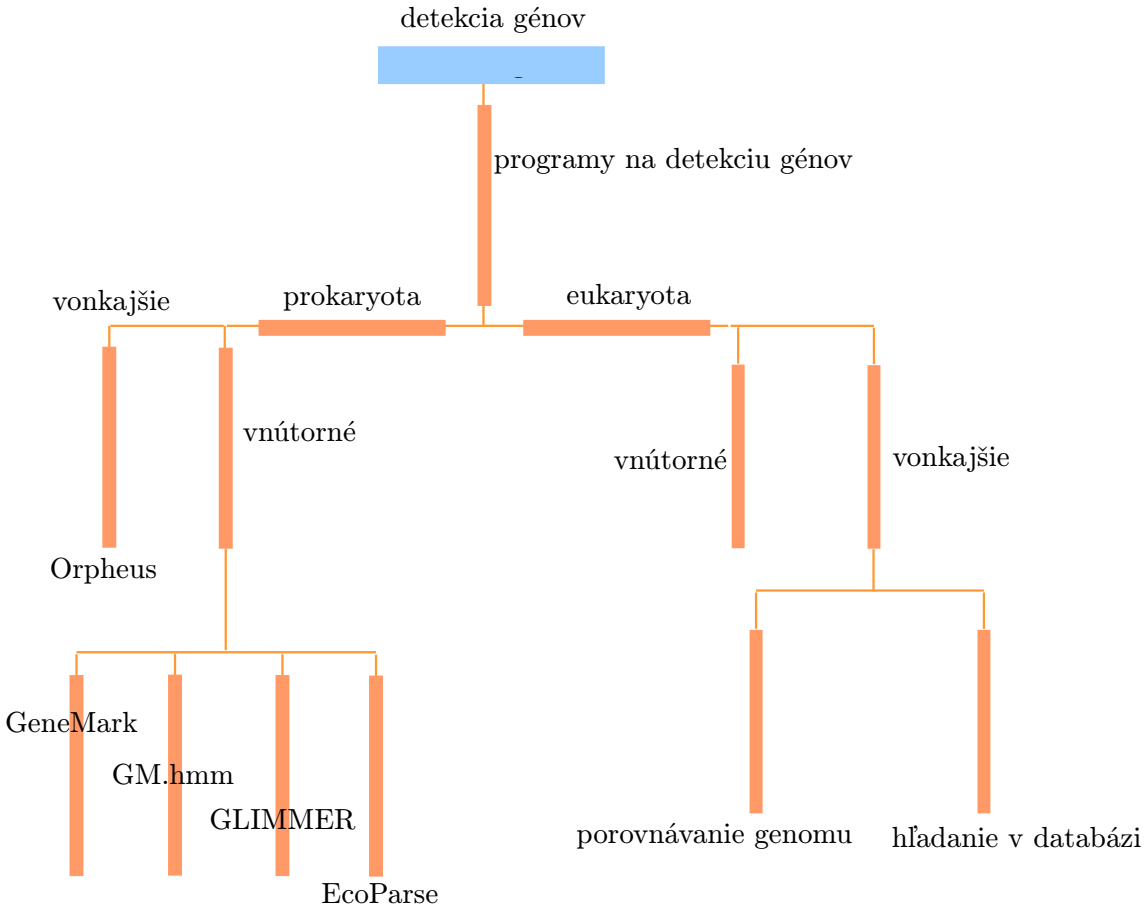


Obrázok 3.2: Prokaryotický promoter

3.4 Algoritmy pre detekciu génov

Metód pre detekciu bakteríalnych génov bolo navrhnutých už veľmi veľa. Pôvodné techniky boli veľmi jednoduché a dlhú dobu obmedzujúce hlavne kôli nedostatočným znalostiam a informáciam o štruktúre a fungovaní DNA a celkovo nedostatočne preskúmanej oblasti molekulárnej biolólogie [11].

Až po nadobudnutí viacerých znalostí a informácií uzreli svetlo sveta zložitejšie a presnejšie metódy a algoritmy s ktorými bolo možné pracovať. Tento popis však nemá za úlohu vysvetliť do podrobna ako dané metódy fungujú, ale oboznámiť s ich existenciou a účelom [11].



Obrázok 3.3: Programy na detekciu génov [11]

3.4.1 GeneMark

Používa nehomogénny Markovov reťazec na reprezentáciu štatistík kódujúcich a nekódujúcich rámcov. Markovov reťazec označuje stochastický proces, ktorý hovorí, že v stave, v ktorom sa proces nachádza je pravdepodobnosť navštívenia ďalších stavov nezávislá na tom, aké stavy boli navštívené. Chovanie je teda bezpamäťové. Markovov reťazec je popísaný vektorom a maticou. Metóda používa dikodónovú štatistiku na identifikáciu kódujúcich regiónov [11].

Táto metóda potrebuje rozsiahlu trénovaciu množinu dát, aby sa dosiahla požadovaná presnosť. Používa sa piateho rádu Markovovho reťazca, kde Markovov model má 4096 možností pravdepodobností okrem pravdepodobnsti výskytu pentametru. Metóda používa ohodnocovanie, ak napríklad snímok obsahuje štart a stop kodón, skóre ktoré dostane je vyššie, dané ORF môže byť gén, za predpokladu, že prejde záverečným testom na prekrývanie génu. Pri teste na prekrývanie sa môžu testované ORF prekrývať len do určitého maximálneho rozsahu [11].

Existuje modifikácia tejto metódy a volá sa GeneMark.hmm. Originálna GeneMark metóda používa štatistiku nukleotidov na lokáciu potencionálnych ORF. V tejto modifikácii je Markovov model transformovaný do skrytého Markovho modelu [11].

Existuje niekoľko aplikácii postavených na tejto metóde. GeneMark je dokumentovaný ako jeden z najpresnejších vyhľadávačov prokaryotických génov. Bol vyvynutý na Inštitúte technologií v Atlante [11].

3.4.2 GLIMMER

Tento algoritmus tak isto používa Markovov model na predikciu génov, ale tento model má trochu inú štruktúru ako model používaný pri GeneMark metóde. Tento model sa nazýva interpolovaný Markovov model. Vyhľadáva dlhé sekvencie ORF, ktoré neprekrývajú iné dlhé ORF. Tieto rámce potom označí ako ORF, ktoré môžu byť s veľkou pravdepodobnosťou reálne gény [11].

3.4.3 Orpheus

Orpheus používa vo svojom programe informácie ktoré ostatné programy ignorujú. Používa vyhľadávanie v databáze, aby mohol stanoviť gény, čiže patrí medzi vonkajšie metódy. Tento počiatočný súbor génov je použitý na definíciu kódovacích štatistík pre organizmus, v tomto prípade sa pracuje na úrovnu kodónov a nie dikodónov, ako pri metódach predtým. Tieto štatistiky sú použité na definovanie dlhšieho súboru potencionálnych ORF [11].

Východzím bodom pre Orpheus je určenie počiatočných najviac dôveryhodných génov podľa homologickej sekvencie na rozpoznávanie známych proteínov. Táto metóda využíva vysledkov získaných z predchádzajúcich metód, čiže pre svoju funkčnosť je závislá na už objavených dátach inými metódami, čiže množstvo hypotetických proteínov z predchádzajúcich projektov boli pridané do databázi, z ktorých Orpheus čerpá. Používa ale len položky databázi, ktoré sú veľmi spoľahlivé. Vyhľadané regióny sú zarovnané na najbližší štart a stop kodón, a tento súbor ORF sekvencií sa použije na stanovenie štatistiky kodónu [11].

Kapitola 4

Technické riešenie

4.1 Použité nástroje a vývojové prostredie

Ako nástroj pre prácu som si zvolil vývojové prostredie Netbeans. Netbeans je úspešný open source projekt vytvorený v programovacom jazyku Java. Umožňuje prehladné písanie a editáciu zdrojového kódu s podporou takmer akéhokoľvek programovacieho jazyka [9].

4.1.1 Webový server

Ako webový HTTP server bol použitý LAMP server (Linux Apache Mysql Php), kde miesto php môže byť použitý iný skriptovací jazyk ako napríklad perl alebo python.

Verzie jednotlivých použitých častí LAMP servera:

- Ubuntu v10.10 linux kernel 2.6.35-28-generic-pae
- Apache/2.2.16
- PHP 5.3.3-1ubuntu9.3

Databáza pri riešení zadania potrebná nebola, pretože systém si nevyžadoval ukladania žiadnych perzistentných dát.

4.1.2 Značkovací jazyk HTML

Jazyk HTML (HyperText Markup Languane) je vyvynutý z univerzálného značkovacieho jazyka SGML. Vývoj tohoto jazyka začal niekedy v roku 1991 a zastal na verzii HTML 4.01, momentálne je vo vývoji očakávana verzia HTML 5.0, ktorá ma priniesť prevratné novinky vo vývoji webových aplikácii kde množstvo funkcionality, ktoré je nutné riešiť inými nástrojmi ako napríklad java-script bude implementovať už HTML samotné. Kôli nekompatibilite a podpore alebo skvôr nepodpore v rôznych prehliadačoch je to zatial len vízia budúcnosti čo sa masového nasadenia a využívania týka. HTML všeobecne slúži na zobrazovanie statických informácii na webe či už v textovej alebo grafickej podobe [6].

Štruktúra HTML dokumentu má svoju predpísanú štruktúru, ktorú je potrebné striktne dodržiavať ak samozrejme chceme aby web bol valídny. O validáciu, dodržiavanie a vydávanie štandardov sa stará organizácia w3c, ktorá poskytuje niekoľko svojich nástrojov priamo online na webe, ktoré umožňujú kontrolu validity podľa užívateľom alebo zdrojom navolených verzií štandardov, podľa ktorých je web napísaný.

Každý dokument by mal obsahovať povinnú direktívu `<!DOCTYPE`, ktorá hovorí o aký typ dokumentu ide. Určite nesmie chýbať koreňový element `<html>`, ktorý reprezentuje celý dokument. Celý HTML dokument sa skladá z dvoch väčších častí, z hlavičky a tela. Hlavička dokumentu obsahuje informácie o stránke, inkludované súbory ako napríklad štýly stránky alebo java skripty. Udáva informácie pre vyhľadávače a definuje titulok pre danú stránok. V tele stránky sa nachádzajú už prvky, ktoré budú zobrazené pri návšteve danej stránky.

4.1.3 Štýlovací jazyk CSS

Štýlovací jazyk CSS (Cascading Style Sheets) v preklade kaskádové štýly. Tieto štýly služia na formátovenie prvkov na stránke napísanej v HTML. Existuje viacej možností ako formátovať dané elementy, či už prostredníctvom samotného HTML alebo použitím CSS štýlov. Veľkou výhodou je možnosť oddeliť formátovanie od samotného elementu a sprehľadniť tým HTML kód.

Aj keď sa jedná pre túto prácu na prvý pohľad možno o nepodstatnú vec, sú to práve CSS štýli, ktoré veľkým sústom pomohli realizácii tejto práce. Vďaka skvelím formátovacím nástrojom a onhover akciám, ktoré CSS podporuje bolo možné zostrojiť výstup programu a zkvalitniť prehľadnosť, čo pri výukovom programe je podstatná záležitosť.

Menším problémom bola ale podpora v niektorých prehliadačoch pretože boli použité prvky štandartu CSS verzie 3, ktoré sa ešte stále netešia jednotnej podpore vo všetkých prehliadačoch.

4.1.4 Klientský jazyk JavaScript

Java script je populárný skriptovací jazyk, ktorý je podporovaný vo všetkých známejších a rozšírenejších internetových prehliadačoch. Pomocou java scriptu je možné programovať rôzne udalosti, ktoré sa vykonajú na počítači u klienta. Hlavným využitím jazyka v tejto práci je prispôsobenie a zkvalitnenie uživatelského rozhrania. Okrem toho java script dokáže volať skripty iných jazykov, predávať im dáta na vstup a čitať výstup z daných skriptov. Nieje problém zavolať php, perl alebo python skript a spracovať požiadavok na pozadí. Tento fakt sa využíva nato aby nemuselo dôjsť k refreshu stránky pri uživateľom vyvolanej akcii, čo vedie k uživatelsky príjemnejšej práci s rozhraním. Táto technológia sa nazýva ajax.

Pre rýchlejšiu prácu a programovanie v java scripte, je využitý open source framework Jquery.

4.1.5 Serverový jazyk PHP

PHP je skriptovací jazyk, ktorý sa dá použiť na tvorbu dynamických webových stránok, čo bol vlastne aj dôvod jeho vzniku v minulosti. V súčasnej dobe je php veľmi rozšírené pretože je veľmi dobre použitelné pre malé a stredné projekty. Skripty sú spracovávané na strane serveru. Klient odošle požiadavok na server, server vykoná určitú činnosť a vráti odpoveď klientovi. Fungovanie a beh php zabezpečuje apache server.

Tento jazyk je možné použiť na mnoho účelov - spracovanie textov, spracovanie grafiky alebo práca so súbormi. Veľkou výhodou je možnosť pracovania s veľkým množstvom databázových systémov ako napríklad MySQL, ORACLE, PostgreSQL a ďalších. Pomocou php je možné komunikovať viacerými protokolmi okrem základného HTTP je možnosť pripojenia cez FTP, SMTP, POP3 alebo LDAP.

Medzi hlavné výhody tohoto jazyka patrí široký repertoár funkcií. V PHP je možné nájsť funkciu takmer na všetko čo patrí medzi bežné programátorské problémy. Samozrejme nechýba podpora regulárnych výrazov a rôznych funkcií pracujúcich s reťazcami, vyhľadávaním v reťazcoch a filtrovaním, čo som veľmi ocenil pri implementácii algoritmov tejto práce. Okrem iného php disponuje aj solídne prepracovanou dokumentáciou a rozsiahlim diskusným fórom kde je možné nájsť riešenie takmer na všetko.

Nevýhodou tohoto jazyka je ale rýchlosť spracovania a rýchlosť výpočtu pri rozsiahlejších vstupoch. Pre túto prácu sú ale tieto obmedzenia zanedbatelné pretože veľkosť vstupov nieje tak rozsiahla aby čas spracovania narastal do neprípustných časov.

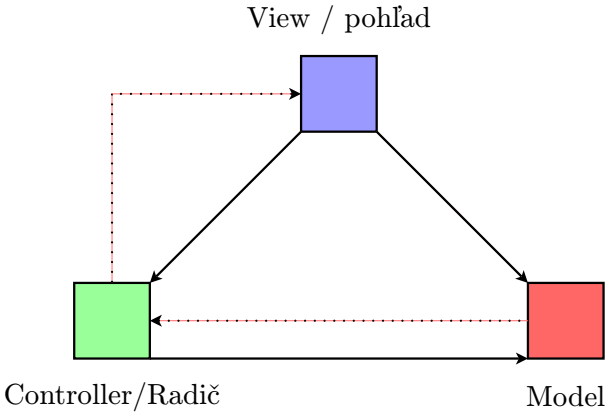
Vytváraná webová aplikácia je z veľkej časti na PHP založená. Valnú väčšinu funkčnosti webu zaisťuje práve táto technológia.

4.2 Nette framework

Nette framework je veľmi silný a efektivný nástroj na tvorbu webových aplikácií založených na PHP 5. Keďže s týmto frameworkom pracujem už dlhšiu dobu bol pre mňa jasnou volbou. Vďaka svojej efektívne navrhnutej objektovo založenej architektúre a vďaka svojim nástrojom pre prácu umožňuje rýchlejší a prehladnejší vývoj webových aplikácií. Jedná sa o český open source projekt, na ktorom sa podielajú autori či už z Česka alebo Slovenska. Silná uživatelská základňa a hlavne domáce prostredie robí z tohoto frameworku veľmi dobre použitelný nástroj [5].

4.2.1 MVP návrhový vzor

MVP (Model View Presenter) je softwarová architektúra, ktorá rozdeľuje aplikáciu do troch vrstiev. Model, charakterizuje ukladanie, spracovanie a načítanie dát napríklad z databázy, alebo z čohokoľvek čo si uživatel zvolí. View (pohľad) slúži na zobrazovanie dát, berie obsah, ktorý mu generuje presenter alebo aj kontroler. Takáto separácia je dôležitá pre udržanie poriadku a prehladnosti pri vývoji webových(a nielen webových) aplikácií. Nette framework je navrhnutý pre písanie priamo MVP aplikácií. Na obrázku 4.1 je zobrazený vzťah medzi prvkami MVP architektúry [5].



Obrázok 4.1: MVP návrhový vzor [5]

Kapitola 5

Implementácia

5.1 Analýza problému a vytýčenie cieľov

Cieľom praktickej časti práce nieje konkurovať zložitým nástrojom na detekciu génov. Dané aplikácie sú vyvýjané za iným zámerom s inými cieľmi a väčšinou na aplikácii a tvorení zložitejších metód pracuje niekoľkočlenný tým. Cieľom práce je demoštrácia predom zvolených prvkov a faktov týkajúcich sa detekcie génov. Profesionálne aplikácie sa zaoberajú vždy väčšinou jednou časťou problematiky a nezahŕňajú v sebe ďalšie prvky úzko spojené s hlavnou myšlienkou, ktorú nesú ako napríklad množstvo aplikácii, ktoré detekujú kódujúce sekvencie, už v sebe nezahŕňajú ďalšie postupy, kvôli ktorým k danej detekcii musí dôjsť. Aplikácia, ktorá vznikla za cielom vyhotovenia praktickej časti tejto práce má za úlohu ako už bolo povedané demonštrovať prenos genetickej informácie z DNA do proteínu a vhodným a prehľadným spôsobom ukázať ako celý tento proces prebieha.

Veľkým problémom pre rozsiahlosť tejto problematiky bolo správne vyhodnotiť a zvoliť, na ktoré prvky sa sústrediť, ktoré je vhodné pokladať za dôležité pre demonštráciu. Keďže ide o výukovú aplikáciu, nieje možné v rámci projektu ako je táto bakalárska práca aplikovať všetky postupy, ktoré sa pri reálnych riešeniach používajú, ide hlavne o algoritmy a časť detekcie kódujúcich sekvencií, čomu by bolo vhodné venovať samostatnú prácu. Predikcia génov je najsložitejšou časťou tejto práce a nielen tej, ale aj všeobecne čo sa bionformatiky týka (predikcia génov u eukaryot), hlavne kvôli potrebe veľmi veľa dát v úlohe trénovacích množín. Preto pre zjednodušenie a lepšie pochopenie sú do programovej časti aplikácie zaradené základné fakty a informácie. Problematika transkripcie, vytvorenie mRNA a následná tvorba polyptidového proteínového reťazca už pri objavenej kódujucej sekvencii nieje problém a je možné v tomto prípade aplikovať pevne dané pravidlá.

5.2 Vyhľadávanie promotora

Ako už bolo zmienené vyhľadávanie kódujúcich sekvencií nieje až tak triviálna záležitosť a pre presné výsledky je potrebné implementovať niekoľko pravidiel aby sa dosiahlo požadovaného výsledku. Preto pre potreby tejto práce je aplikácia obmedzená čo sa týka promotoru len na vyhľadávanie tata boxu a Gilbertovho boxu. Čo sa implementačného hľadiska týka všetky algoritmy sú aplikované na vstupnú sekvenciu zadanú uživateľom. Pracuje so základným pracovným DNA vláknom v smere 5´- 3´.

5.2.1 Vyhľadávanie Tata boxu

Pri vyhľadávaní tata boxu program prechádza vstupnú sekvenciu a hľadá nálezy tata boxu, čiže sekvencie TATAAA. V praxi by sa mali brať do úvahy aj iné nukleotidy, pretože v tejto sekvencii sa môžu nachádzať aj iné bázy, každá s určitou pravdepodobnosťou v závislosti na operóne.

Po vyhľadaní všetkých takýchto sekvencií, prichádza narad kontrola ďalšieho pravidla. V praxi od každého TATAAA boxu vo vzdialenosti približne 150 bázi sa nachádza štart kodón, v tejto aplikácii sa program obmedzuje na interval 20 až 30 nukleotidov. Kde akákoľvek iná vzdialenosť sa nebere do úvahy a vyhodnocuje ako štart kodón ktorý nepatrí k danému tata boxu. Týmto spôsobom sa vyfiltrujú všetky nálezy boxov, ktoré neoznačujú žiadnu kódujúcu sekvenciu, pretože sa tam nenachádza žiadny štart kodón. Ak sa nájde nejaký štart kodón algoritmus pokračuje ďalej a vyhľadáva sa jeho patričný koniec ak sa koniec nenájde nikde do konca reťazca situácia sa vyhodnotí tak, že daný box a jeho štart kodón neoznačujú žiadnu kódujúcu sekvenciu, pretože v našej aplikácii sa vyžaduje demonštrácia translácie a transkripcie a bez konca reťazca neni možné transláciu vykonať.

Jeden tata box môže byť spoločný pre viac génov, tie sa môžu prekrývať. Záleži od posunutia od nálezu tata boxu, keďže aminokyselinu kóduje jeden triplet.

V praxi by promotor mal obsahovať aj Gilbertov box, aplikácia ale akceptuje aj výskyt len tata boxu, je na uživateľovi či si zvolí aby program detekoval aj Gilbertov box.

Na konci spracovania sú k dispozícii všetky sekvencie, ktoré splnili dané pravidlá, čiže sú odfiltrované tata boxi a k ním príslušné ORF.

- CGTTTAC **TATAAA** (20-30 báz) **ATG** .... **TGA** (Valídna sekvencia)
- AACCTTA **TATAAA** (50 báz) ATG ... TGA (Nevalídna sekvencia, nesplňuje pravidlo)
- AACTGAA **TATAAA** CCC AAA TTT AAA AAA GGG CCC **ATG** AC**A** **TG**A GGC CCG **TGA** C**TA** **G**AA (box spoločný pre 2 ORF)

5.2.2 Vyhľadávanie Gilbertovho boxu

K vyhľadávaniu Gilbertovho boxu dochádza až po časti kde sa vyhľadáva tata box, pretože ako už bolo spomenuté program akceptuje možnosť existencie promotoru aj bez prítomnosti Gilbertovho boxu.

Algoritmus sa aplikuje na sekvencie kde sa našiel tata box. Vyhľadáva nukleotidy v tvare TTGACA vo vzdialenosti 16 až 18 báz pred tata boxom. Tak isto ako pri detekcii tata boxu v praxi tento box nemusí mať striktne tento tvar, pre možnosť jednoduchšej demonštrácie je tento fakt v implementácii vynechaný a pracuje sa len so základným tvarom sekvencie boxu.

- CGTTTAC **TTGACA** (16-18 báz) **TATAAA** .... **atg** (Valídna sekvencia)
- CGTTTAC **TTGACA** (menej alebo viac ako 16-18) **TATAAA** (Nevalídna sekvencia)

5.3 Transkripcia, translácia a prevod z proteínu na RNA a DNA

Pri vykonávaní translácie program spracuje výstupy a výsledky z prechdádzajúcej časti, čiže nájdené potencionálne kódujúce sekvencie. začína na +10 offsete od nálezu tata boxu až do nálezu stop kodónu a to pre každé príslušné ORF pre daný box. Z tejto sekvencie vyvtvorí mRNA vlákno. Dáta zpracúvava z pôvodného pracovného DNA vlákna kde nahrádza len thymín za uracil. V reálnom živote sa ale tento jav deje na druhom templátovom vlákne, pre výpočet je ale jednoduchšie použiť pracovné vlákno, ušetrí sa tým práca s vytváraním komplementu k vytvorenej mRNA z templátového vlákna.

Translácia sa vykonáva z vytvorenej mRNA, kde od štart kodónu pre jednotlivé ORF je generovaný proteinový reťazec. Podľa tripletov sú z tabulky proteínov vyberané príslušné aminokyseliny a tým sa zostavuje výsledný reťazec.

Ďalšou zložkou práce je implementovanie opačného prevodu a to podľa proteínového reťazca vytvoriť príslušné RNA a DNA. V reálnom biologickom živote niečo takéto nieje možné, prenos informácie sa vykonáva len smerom z DNA do proteínu. V tejto časti je dôležité pripomenúť, že genetický kód je degenerovaný, danú aminokyselinu kóduje dva a viac kodónov, okrem výnimiek ako napríklad metionín, ktorý je kódovaný jedným kodónom, preto aj spätné zostavovanie sekvencie je veľmi nepresné keďže konkrétny proteínový reťazec môže byť tvorený obrovským množstvom možností zoskupenia kodónov.
Pre proteínový reťazec Metionin - Histidin - Glutamin - Stop sú možné nasledujúce kombinácie:

- AUG - CAU - CAA - UAA
- AUG - CAA - CAA - UAA
- AUG - CAU - CAG - UAA
- AUG - CAA - CAG - UAA
- AUG - CAU - CAA - UGA
- AUG - CAA - CAA - UGA
- AUG - CAU - CAG - UGA
- AUG - CAA - CAG - UGA
- AUG - CAU - CAA - UAG
- AUG - CAA - CAA - UAG
- AUG - CAU - CAG - UAG
- AUG - CAA - CAG - UAG

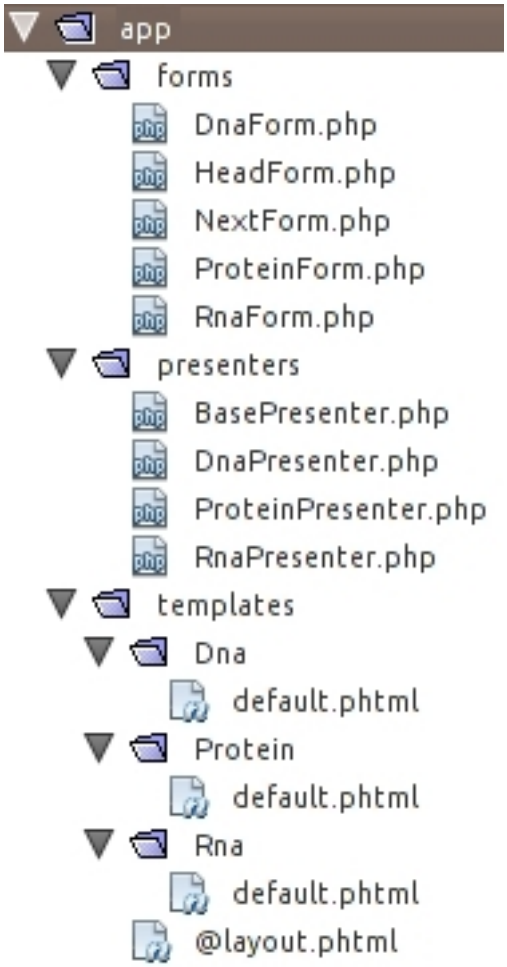
Z príkladu je možné vidieť koľko rôznych kombinácii je možné vytvoriť a to sa jedná len o 2 aminokyseliny ak neni počítaný štart a stop kodón. Aplikácia výhodnotí len jeden demonštračný výsledok, takzvaný prvý príslušný možný nález ako by daná sekvencia mohla vyzerať.

5.4 Zložky praktickej časti práce

V tejto časti je popísaná adresárová štruktúra a jej obsah a uživatelské rozhranie.

5.4.1 Adresárová štruktúra

Adresárová štruktúra je navrhnutá podľa vzoru štruktúry, s ktorou pracuje nette framework. Štruktúra ale môže byť ľubovoľná, stačí načítať jadro frameworku nastaviť mu skenovacie adresáre a on následne načíta všetky súbory respektíve triedy do hlavného programu. Dôležitým adresárom je adresár `documet_root` (obrázok 5.2), ktorý obsahuje všetky potrebné adresáre a súbory pre zobrazenie stránky, nachádza sa tu aj hlavný index súbor. Do tohoto adresára je nutné pri inštaľácii aplikácie namapovať cestu virtuálneho hosta web servera.

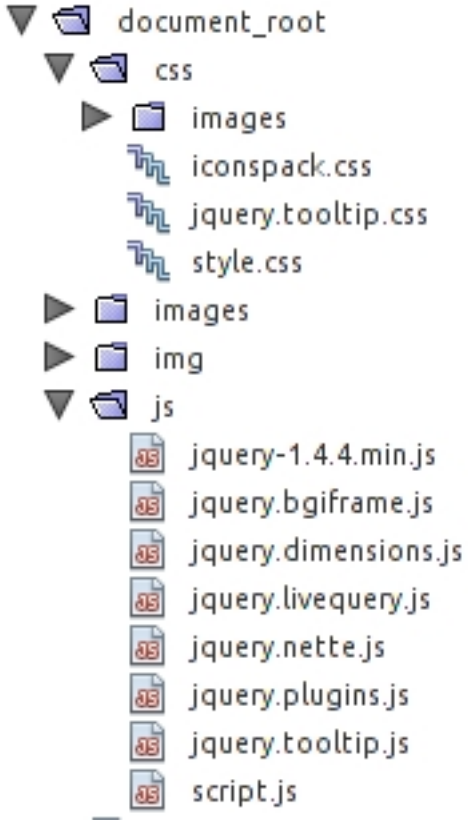


Obrázok 5.1: Aplikačná adresárová štruktúra z vývojového prostredia NetBeans

Súbory spracúvávajúce prevod z DNA do proteínu:

- app/presenters/DnaPresenter.php

  Tento súbor obsahuje funkcionalitu potrebnú pre prenos informácií z DNA do proteínu. Zahŕňa v sebe implementáciu agloritmov na vyhľadávanie štartovacích signálov, vyhľadávanie tata boxu a Gilbertovho boxu, hľadanie štart a stop kodónov. Vyko-



Obrázok 5.2: Adresárová štruktúra z vývojového prostredia NetBeans

náva transláciu a transkripciu z uživatelom zvolených sekvencií. Spracované dáta sa vypisujú v samostatnom súbore popísanom nižšie.

- app/forms/DnaForm.php sapp/templates/Dna/default.phtml

  Súbor ktorý vypisuje dáta z DnaPresenter-a, využíva nástroje nette frameworku a s použitím css štýlov a java scriptu formátuje výstup do uživatelsky prijatelnej podoby.

Súbory spracúvávajúce prevod z RNA do proteínu:

- app/presenters/RnaPresenter.php

  Z implementačného hľadiska je tento súbor ochudobnený o vyhľadávanie štartovacích signálov keďže mRNA sekevenciu zadáva uživatel. Skript obsahuje algoritmy na vyhľadávanie štart a stop kodónov a vykonáva transkripciu zo zvolených ORF.

- app/templates/Rna/default.phtml

  Súbor tak isto formátuje dáta spracované z RnaPresentera.

Súbory spracúvávajúce prevod z proteínu do RNA a DNA:

- app/presenters/ProteinPresenter.php

  Tento skript implementuje prevod z proteínového reťazca naspäť na mRNA a DNA. Aplikuje len jednoduchý prevod na základe vyberania z poľa podľa klúča. Nepočíta

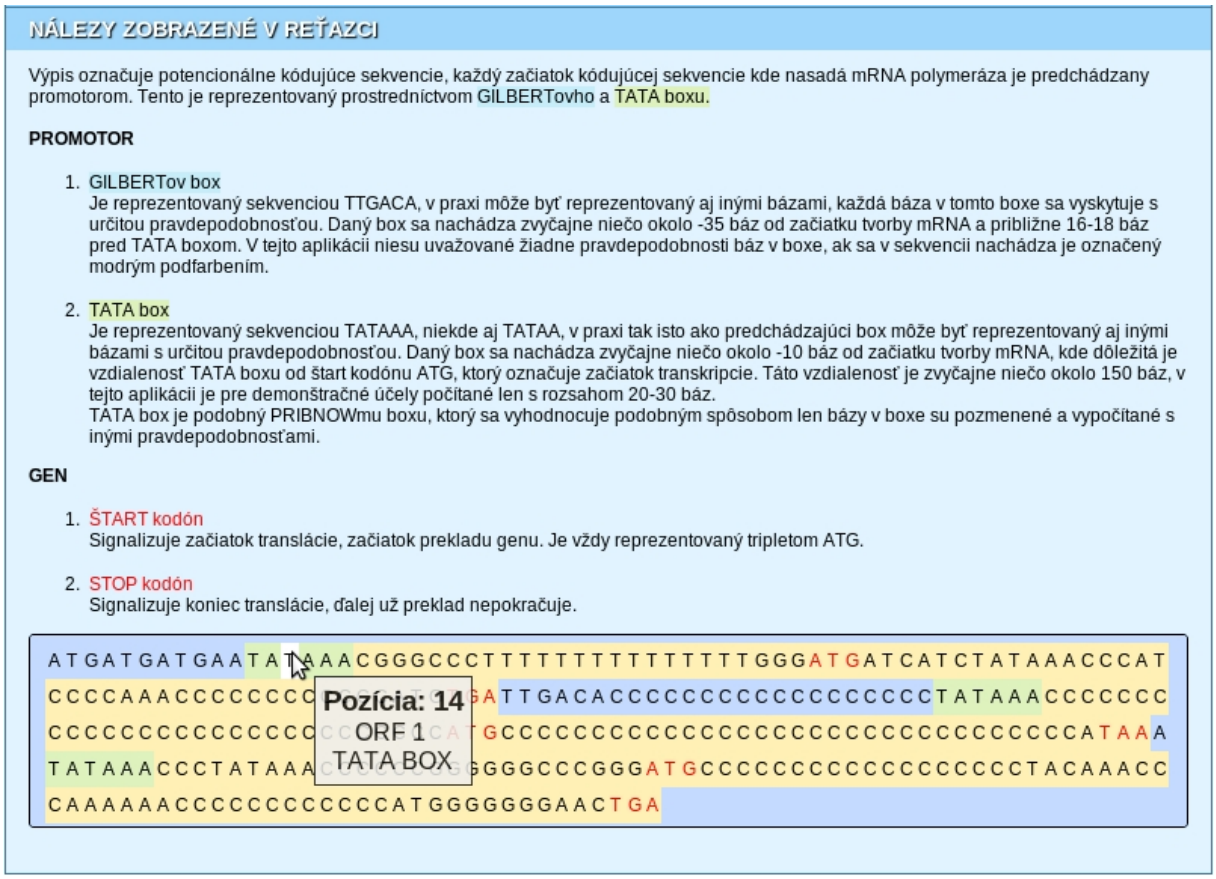
s viacerými možnosťami možných sekvencii, čo kôli degenerovanosti genetického kódu umožňuje vytvoriť obrovské množstvo kombínacií.

- app/templates/Protein/default.phtml

  Naformátované dáta pre uživatela sú výstupom tohoto súboru.

5.4.2 Uživatelské rozhranie

Počas práce bolo snahou uspôsobiť uživatelské rozhranie k čo najväčšej prehľadnosti a jednoduchosti ovládania. Bolo dôležité vhodne zvoliť formát výstupu respektíve výpisu získaných dát a štatistík. Uživatel si môže v hlavnom menu v hlavičke stránky vybrať jednu z troch možností.



Obrázok 5.3: Nálezy zobrazené v reťazci

Možnosti navigácie:

- DNA - RNA - PROTEIN
- RNA - PROTEIN
- PROTEIN - RNA - DNA

Pri prevode z DNA cez RNA do proteínu, má uživatel na výber z dvoch možností. Vstup môže zadať buďto zadaním vlastnej vstupnej sekvencie, alebo má na výber z možností vzorových demonštračných sekvencií. Po navolení daného vstupu program dáta spracuje a vytlačí na výstup v požadovanej forme.

Pri prvom spracovaní sa uživateľovi vytlačia priebežné medzivýsledky:

- 10 najpravdepodobnejších nálezov (Obrázok 5.4)
- Zobrazenie celého reťazca (Obrázok 5.5)
- Zobrazenie nálezov v reťazci (Obrázok 5.3)

Pri každom výpise je zobrazené pri prejdení myši pre lepšiu orientáciu poradové číslo nukleotidu v reťazci. Pokial je to potrebné pre lepšie pochopenie a prehľadnosť okrem pozície sú vypisované aj iné informácie, napríklad o akú bázu ide, či je to štart alebo stop kodón alebo či ide o nejaký štartovací signál pre transkripciu.

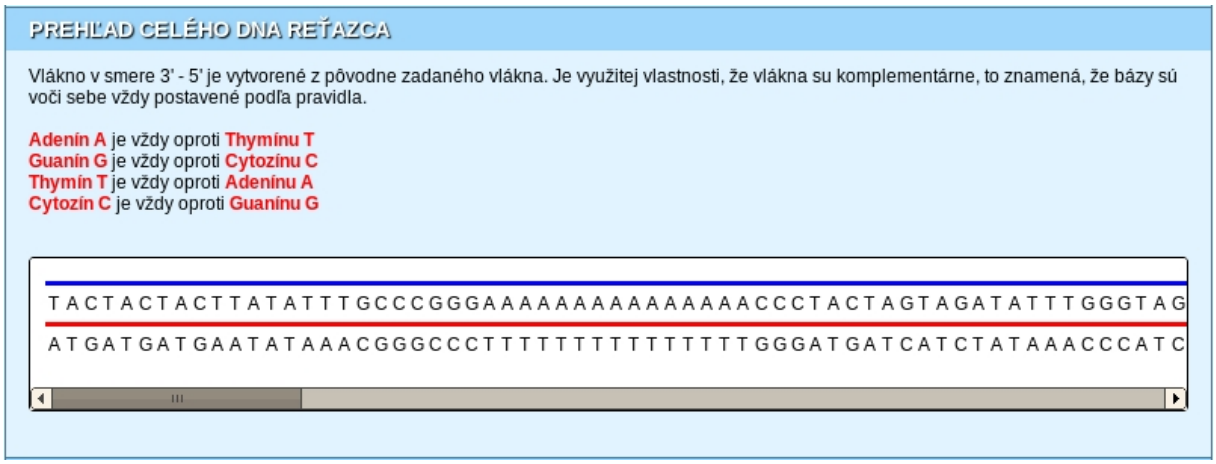
Obrázok 5.4: Výpis 10 najpravdepodobnejších sekvencií

Z výpisu 10 najpravdepodobnejších sekvencií si uživateľ môže vybrať tie, ktoré chce použiť ako dátový vstup pre transkripciu a transláciu. Translácia a transkripcia sa vypisuje podobným spôsobom ako predchádzajúce sekvencie, každá báza ma svoj popisok, ktorý sa vypisuje pri prejdení myšou.

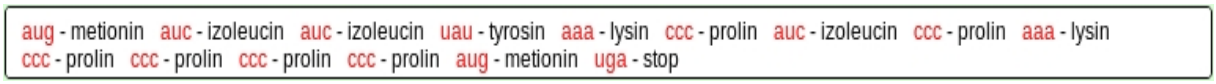
Výsledky spracovania ďalšej časti dát:

- Transkripcia mRNA
- Translácia - textový výpis (Obrázok 5.6)
- Translácia - grafický výpis (Obrázok 5.7)

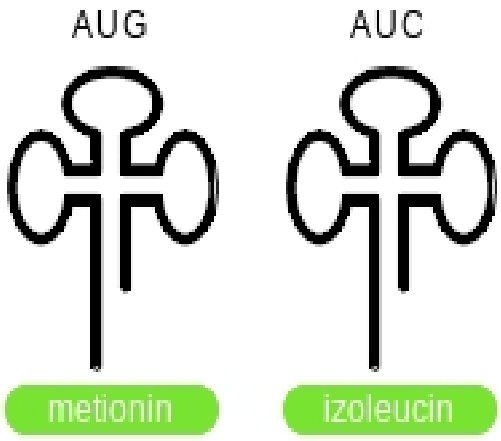
Pri transkripcii sa vypisuje časť pôvodného ORF v obi dvoch smeroch. Translácia ma dva typy výpisov, textový (Obrázok 5.6) a grafický (Obrázok 5.7).



Obrázok 5.5: Výpis celého reťazca



Obrázok 5.6: Textové zobrazenie proteínového reťazca



Obrázok 5.7: Grafické zobrazenie proteínového reťazca

Kapitola 6

Záver

Bakalárska práca sa týka popisu a implementácie problematiky prenosu informácie z DNA pomocou genetického kódu. Na pochopenie všetkých súvislostí danej témy je potrebný aspoň minimálny základ znalostí o DNA, RNA a molekulárnej biológii. Problém pri riešení danej práce spočíval hlavne v teoretickej rovine pochopenia biologických informácií a hlavne správnosti danej teórie keďže ide o problematiku rôzne vysvetlovanú a rôzne aplikovatelnú kvôli svojej všeobecnej a praktickej rozsiahlosti. To obnášalo predovšetkým podrobné naštudovanie netriviálnej úlohy detekcie štartovacích signálov a jeho aplikovanie. Pri riešení tohoto problému pomohli hlavne rady a konzulácie s vedúcim práce a pomerne kvalitná literátura a zdroje na internete.

Cieľom bakalárskej práce bolo teda vyhotoviť výukový systém - webovú aplikáciu, ktorá by demonštrovala základné postupy pri prenose informácie z DNA cez RNA do proteínu a opačne. Za programovací jazyk som si zvolil PHP, ktorý aj napriek svojmu nie veľmi vysokému výpočtovému výkonu pre danú aplikáciu postačoval a jeho využitie bolo výhodne najmä vďaka svojej použitelnosti a rozšíritelnosti v oblasti vývoja webových aplikácií.

Pri vývoji som sa zameriaval nato aby aplikácia bola použitelná, jednoduchá na ovládanie, prehľadná a aby výstižne demonštrovala danú problematiku. Preto za účelom lepšej prehladnosti má výsledná aplikácia obmedzený počet vstupných sekvencií, zjednodušené podmienky výpočtov a detekcií a vynechané niektoré metódy a pravidlá používané na detekciu kódujúcich oblastí. Aplikácia v konečnom dôsledku vyhľadáva a označuje Gilbertov box, tata box a vyhodnocuje k nim príslušné ORF.

Funkčnosť programu bola testovaná a kontrolovaná ručne na prednastavených dátach, keďže sa jedná o výukový systém s upravenými pravidlami v rámci zadania, nebolo možné testovať daný program na reálnych dátach, pretože reálne systémy riešiace podobný problém sú založené na inom princípe za inými účelmi.

Práca bola pre mňa obrovským prínosom a hlavne spestrením informatického štúdia o ďalšiu vednú disciplínu, ktorá s informatikou na prvý pohľad nemá nič spoločné. Pri štúdiu danej problematiky som nadobudol veľa cenných skúseností a nových informácií v oblasti biológie a genetiky.

Výsledná práca by sa dala do budúcnosti obohatiť o prensejšie detekcie boxov v promotore na základe pravdepodobností báz alebo pridaním nejakej métody na detekciu Shine-Delgarnových sekvencií. Ďalším vylepšením by mohlo byť pridanie nejakých štatistických metód alebo nejakých algoritmov pre detekciu génov.

Literatúra

[1] Dna helix [online]. [cit. 2011-04-18].
URL <http://www.wellcome.ac.uk/en/fourplus/DNA.html>

[2] Astrachan, O. L.: APT, Promotion and tata boxes I [online]. [rev. 2007-11-08] [cit. 2011-04-19].
URL <http://www.cs.duke.edu/csed/algoprobs/dna5-5.html>

[3] Benešová, M.; Satrapová, H.: Zmaturuj z chémie. 2002, ISBN 80-7358-030-6.

[4] Churý, L.: Bioinformatika I - Historie a zaměření bioinformatiky,Struktura a funkce DNA,Geny, genomy,buňky,DNA [online]. [rev. 2006-03-05] [cit. 2011-04-18].
URL
<http://programujte.com/?akce=clanek&cl=2006030301-bioinformatika-i>

[5] Grudl, D.: Model-View-Presenter (MVP) [online]. [cit. 2011-04-22].
URL <http://doc.nette.org/cs/model-view-presenter>

[6] Janovský, D.: HTML příručka [online]. [rev. 2011-03-31] [cit. 2011-04-19].
URL <http://www.jakpsatweb.cz/html/>

[7] Kočárek, E.: *Genetika*. Scientia, 2008, ISBN 978-80-86960-36-4.

[8] Martínek, T.: Bioinformatika - Rozpoznávaní genů In: BIF [online]. VUT Fakulta informačných technológií, 2011 [cit. 2011-04-19].

[9] Netbeans: Vítejte u NetBeans a na stránkách www.netbeans.org [online]. [cit. 2011-04-22].
URL <http://netbeans.org/index_cs.html>

[10] Vondrášek, J.; Pačes, J.: Bioinformatika - Predikce genů, tvorba fylogenetických stromů [online]. 2000 [cit. 2011-04-20].
URL <http://bio.img.cas.cz/PfUK2000/110600/>

[11] Zvelebil, M.; Baum, J. O.: *Understanding Bioinformatics*. Garland Science, 2008, ISBN 0-8153-4024-9.

[12] Čársky, J.; Kopřiva, J.; Krištofová, V.: *Chémia pre tretí ročník gymnázií*. Slovenské Pedagogické Nakladateľstvo, 2004, ISBN 80-10-00593-2.

Dodatok A

Obsah CD

- Zdrojový kód aplikácie, php scripty, java scripty, html a css kód
- Elektronická podoba bakalárskej práce
- Latexové zdroje práce
- Inštalačné súbory web servera pre linux a pre windows

Dodatok B

Manuál

B.1 Návod na inštaláciu

B.1.1 Inštalácia web servera

Prvým krokom pri inštalácii práce, je stiahnutie a rozchodenie lokálneho web servera. Návod popisuje kroky pre windows a linuxové distribúcie.

Ak vám nevyhovuje žiadny zo zmienených serverových balíkov, môžete si nainštalovať čokoľvek čo bude mať v sebe apache 2.2.X a podporu PHP 5.3.X. Na iných platformách ako tu popísaných postupujte podľa pokynov z tretích strán určených pre váš operačný systém.

B.1.2 Linux

Pre linuxové distribúcie je možné použiť akýkoľvek dostupný balík s webovým serverom. Manuál sa zaoberá inštaláciou xampp web servera.

1. Otvorte shell a prihláste sa ako root, môžete použiť príkaz su alebo ak patríte do skupiny root uživateľov napíšte sudo -i
2. Rozbalte archiv z `intall/xampp/linux` do adresára `/opt`
   `tar xvfz xampp-linux-1.7.4.tar.gz -C /opt`
3. Server je teraz nainštalovaný v adresári `/opt/lampp`, server spustite jednoduchým príkazom `/opt/lampp/lampp start`
4. Otestujte funkčnosť servera, otvorte adresu http://localhost vo vašom prehliadači, mala by sa zobraziť uvítacia obrazovka xampp webservera

B.1.3 Windows

Existuje veľa webserverov pre platformu windows, odporúčam balík Xampp alebo Wamp, kde prvý menovaný môžete stiahnuť taktiež pre linuxové distribúcie.

1. Stiahnite inštaláciu zo stránky http://www.apachefriends.org/download.php?xampp-win32-1.7.4-VC6-installer.exe alebo z inštalačného CD zo zložky `install/xampp/win/` poprípade z akéhokoľvek iného zdroja.
2. Spustite inštalaláciu a nainštalujte server
3. Spustite xampp control panel a spustite apache server

B.1.4 Vytvorenie virtuálneho hosta a nakopírovanie aplikácie

Linux:

1. Otvorte súbor `/etc/hosts` v nejakom textovom editore
2. Vytvorte nový záznam `127.0.0.1 bio`
3. Otvorte konfiguračný súbor vašeho apache servera pre vytvorenie virtuálneho hosta, väčšinou je to súbor `httpd.conf` vhost alebo site.default, záleží od druhu servera. Pre xampp otvorte adresár `/opt/lampp/etc/httpd.conf`
4. Vytvorte nový záznam podľa príkladu

```
<VirtualHost *:80>
    DocumentRoot "/var/www/bio/document_root"
    ServerName bio
    DirectoryIndex index.php index.html index.html index.htm index.shtml
</VirtualHost>
```

5. Vytvorte adresárovú štruktúru `bio/document_root` v adresári odkial načítava apache zdroj stránok, definovaný väčšinou v `httpd.conf`. (/var/www/bio/document_root)
6. Zkopírujte obsah adresára `application/` z inštalačného DVD do adresára `/var/bio/`
7. Nastavte práva na zápis (777) pre adresáre temp a log
8. Reštartujte apache `/opt/lampp/lampp restart` pre xampp webserver.
9. Otvorte prehliadač a napíšte adresu http://bio

Windows:

1. Otvorte súbor `/windows/System32/drivers/etc/.hosts` (pre windows 7) v nejakom textovom editore
2. Vytvorte nový záznam `127.0.0.1 bio`
3. Otvorte konfiguračný súbor vašeho apache servera pre vytvorenie virtuálneho hosta, väčšinou je to súbor `httpd.conf` vhost alebo site.default, záleží od druhu servera. Pre xampp otvorte adresár `/xamp/apache/conf/extra/httpd-vhosts.conf`
4. Vytvorte nový záznam podľa príkladu

```
<VirtualHost *:80>
    DocumentRoot "C:/xamp/htdocs/bio/document_root"
    ServerName bio
    DirectoryIndex index.php index.html index.html index.htm index.shtml
</VirtualHost>
```

5. Vytvorte adresárovú štruktúru `bio/document_root` v adresári xampp/htdocs
6. Zkopírujte obsah adresára `application/` z inštalačného DVD do adresára `/var/bio/`
7. Reštartujte apache v control paneli xampp servera.
8. Otvorte prehliadač a napíšte adresu http://bio

B.2 Návod na obsluhu

B.2.1 DNA - RNA - PROTEIN

Na hlavnej stránke je umožnené vybrať si z predvolených vzorových príkladov (vždy len jeden), alebo zadať vstupnú sekvenciu. Aplikácia akceptuje sekvencie do 1000 znakov. Pre konečné zobrazenie výsledku uživateľovy stačia 4 kliknutia myšov, takže ovládanie je intuitívne a jasné z vypísaného kontextu. Pri zaklikávaní sekvencii, ktoré chcete vybrať je potrebné klikať na časť elementu okrem radio buttonu (zaškrtávací krúžok).

Pre podrobnejšie zobrazovanie štatistík je potrebné kliknúť na nadpis štatistiky, ktorá sa následne rozroluje.

B.2.2 RNA - PROTEIN

Na tejto stránke je umožnené vybrať si z predvolených vzorových príkladov (vždy len jeden), alebo zadať vstupnú sekvenciu. Aplikácia akceptuje sekvencie do 1000 znakov. Pre konečné zobrazenie výsledku uživateľovy stačia 4 kliknutia myšov, takže ovládanie je intuitívne a jasné z vypísaného kontextu. Pri zaklikávaní sekvencii, ktoré chcete vybrať je potrebné klikať na časť elementu okrem radio buttonu (zaškrtávací krúžok).

Pre podrobnejšie zobrazovanie štatistík je potrebné kliknúť na nadpis štatistiky, ktorá sa následne rozroluje.

B.2.3 PROTEIN - RNA - DNA

Po kliknutí na položku menu, ktorá nesie názov totožný s týmto nadpisom môžete vytvoriť jednoduchý proteínový reťazec pomocou tlačítok kde každé jedno reprezentuje jednu aminokyselinu z tabuľky 20-tich aminokyselín. Reťazec musí začínať metioninom, čiže štartovacím kodónom a musí končiť stop kodónom. Stop sekvenciu neni možné umiestňovať do začatého reťazca inde než na koniec. Metionín je možné umiestniť kdekoľvek do reťazca.

Aplikácia reaguje na uživatelský vstup a koriguje chyby prostredníctvom hlášok pri odosielaní vstupných dát. Pre podrobnejšie zobrazovanie štatistík je potrebné kliknúť na nadpis štatistiky, ktorá sa následne rozroluje.